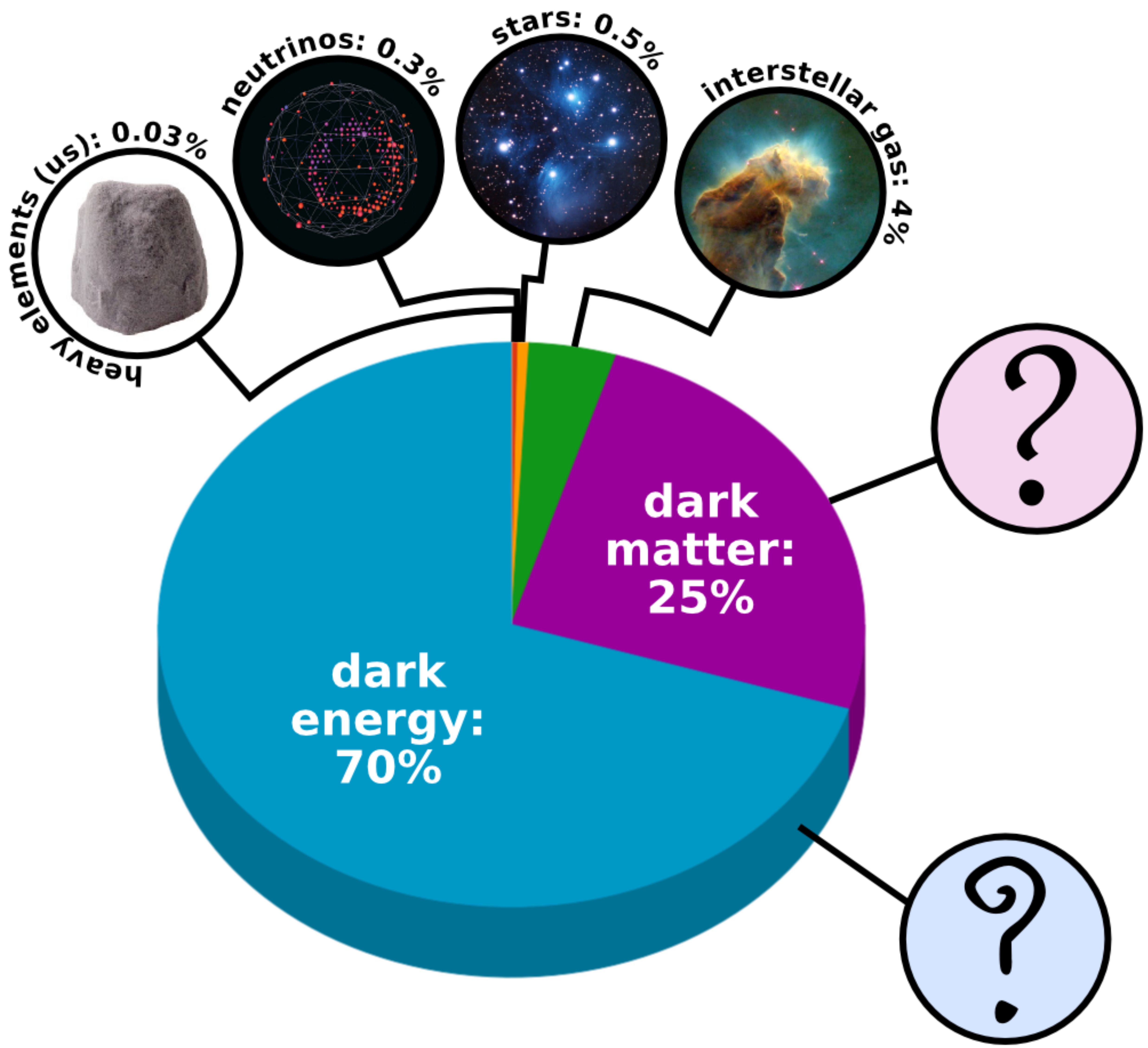
HEP Data Management Introduction - The current status

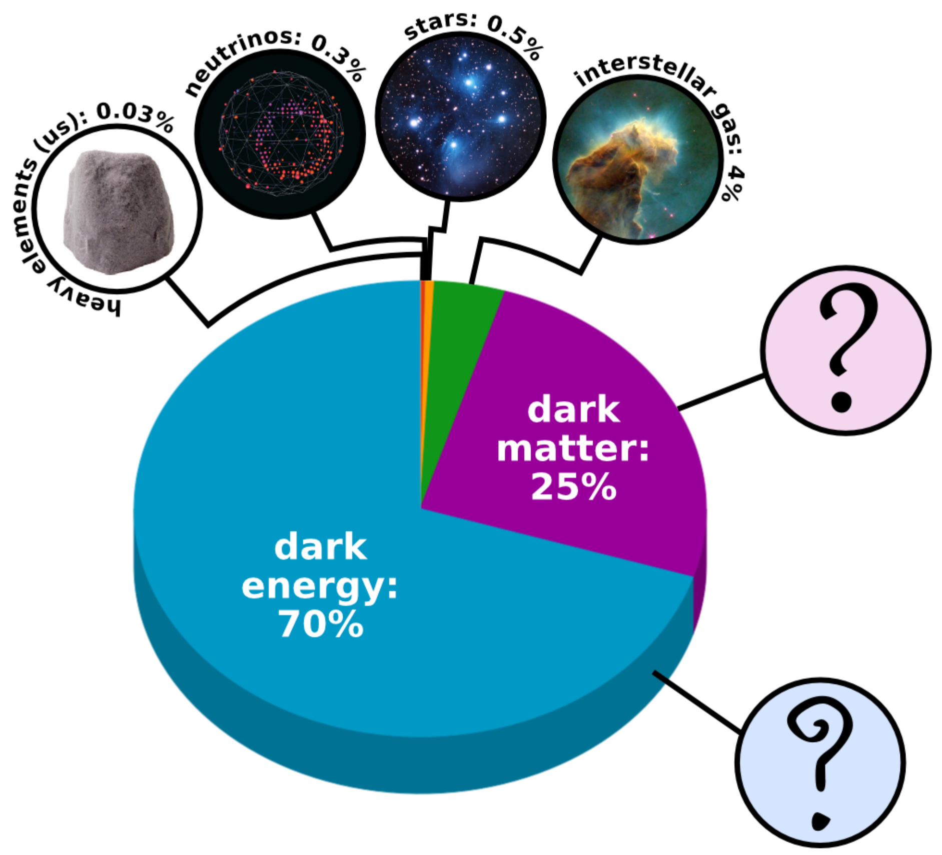
Oliver Gutsche

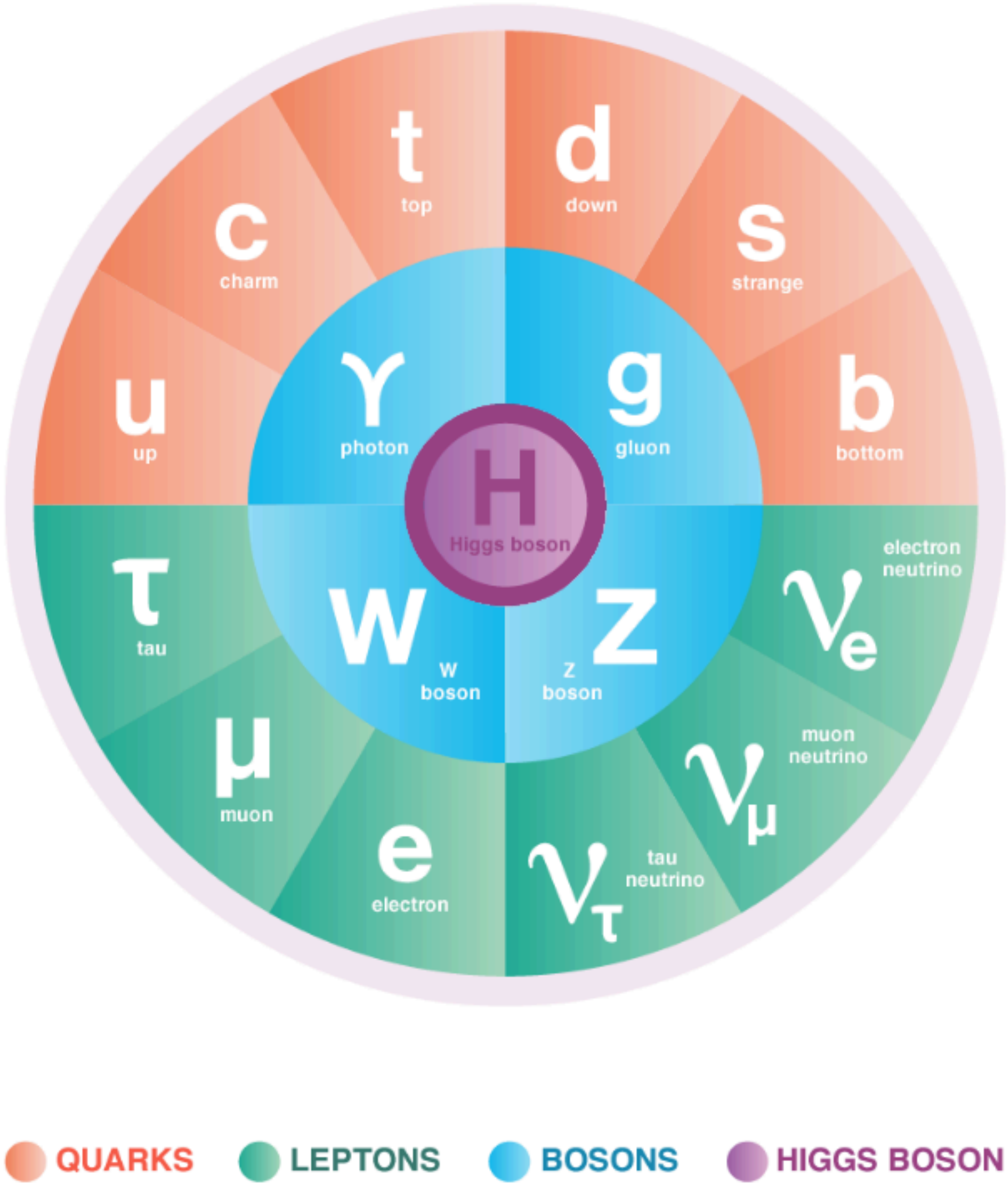
Data Organisation, Management and Access (DOMA) in Astronomy, Genomics and High Energy Physics16. November 2017

Particle Physics

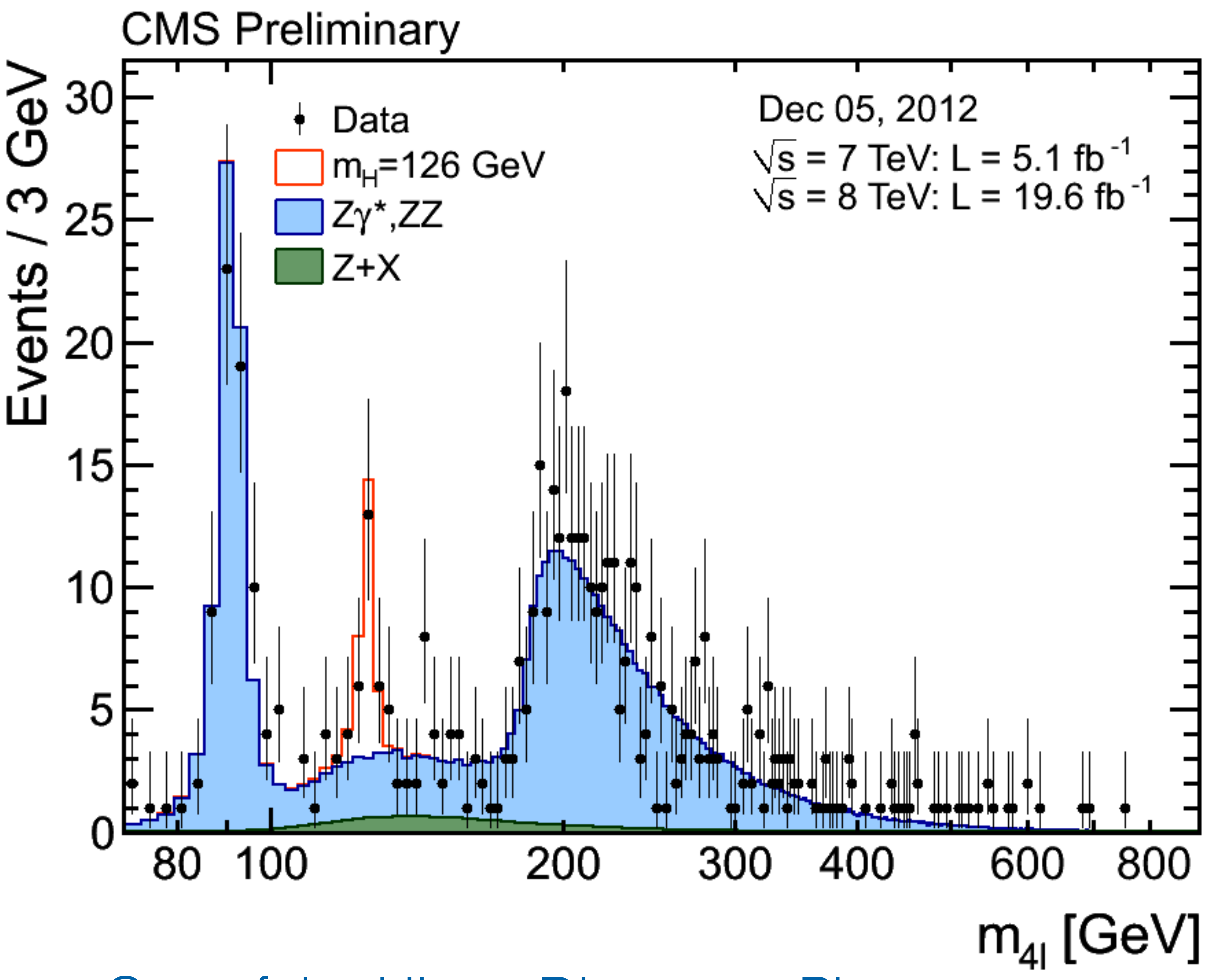


Particle Physics





Experimental Particle Physics

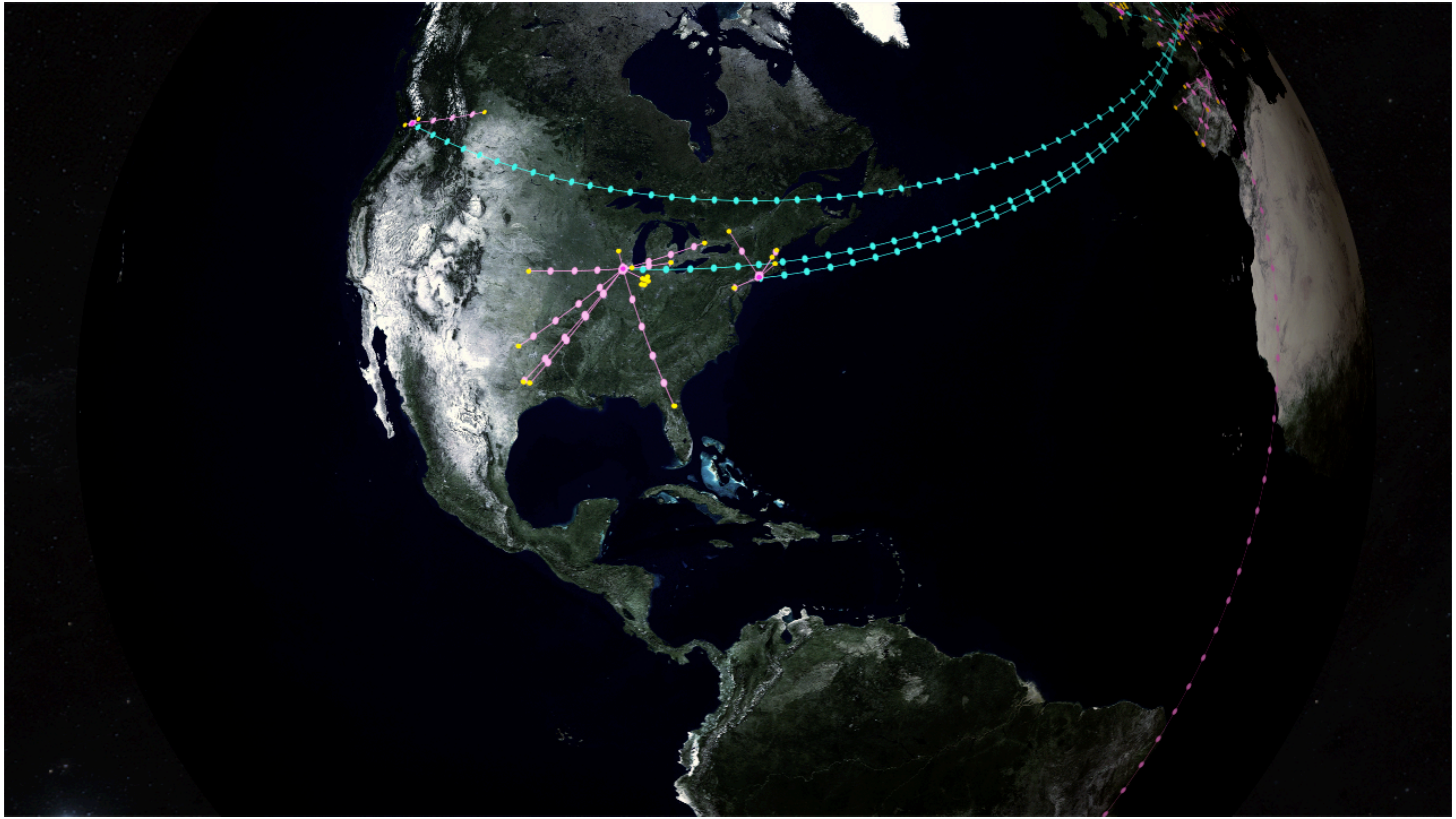


One of the Higgs Discovery Plot:

Higgs Simulation is RED

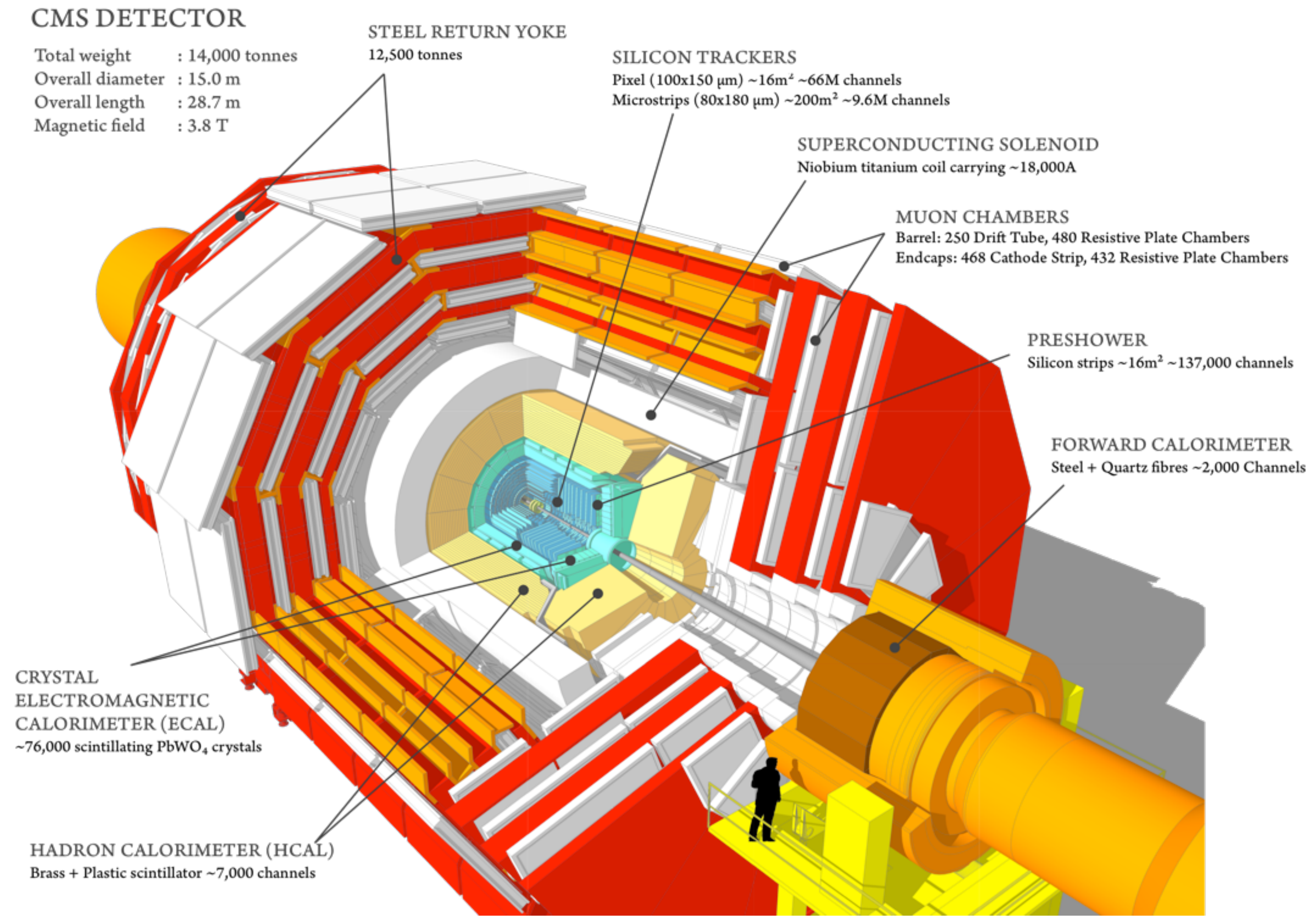
- Detect particle interactions and compare to Standard Model
  - Black dots: measurement
  - Blue shape: simulation of Standard Model
  - Red shape: simulation of new theory (in this case the Higgs)

Introduction Movie



Introduction movie to LHC/CMS computing: http://cds.cern.ch/record/1541893?ln=en

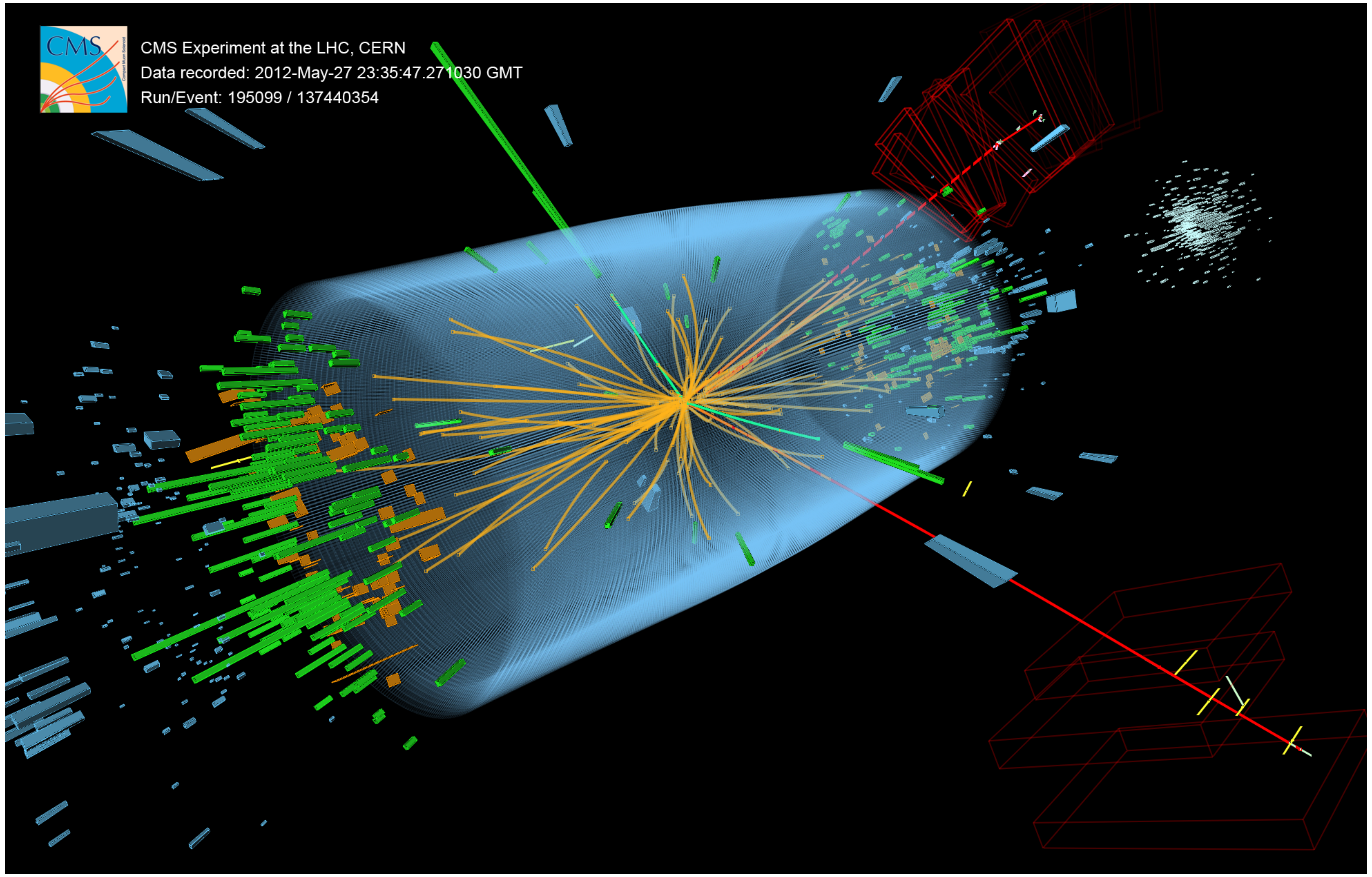
Detector Example - Compact Muon Solenoid (CMS)



- Detector built around collision point
  - One of four detectors at the LHC
- Records flight path and energy of all particles produced in a collision
- 100 Million individual measurements (channels)
  - Grouped by detector component
- All measurements of a collision together are called: **event**

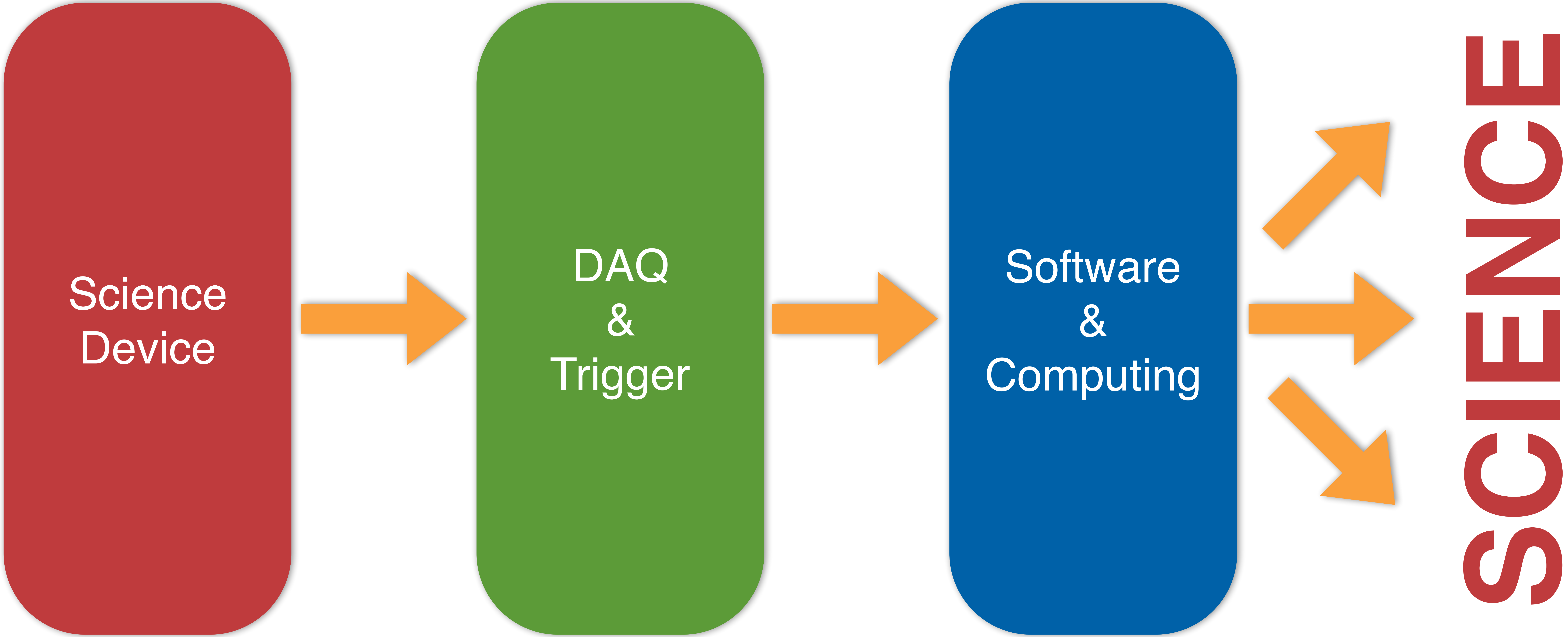
Fermilab

Science Potential

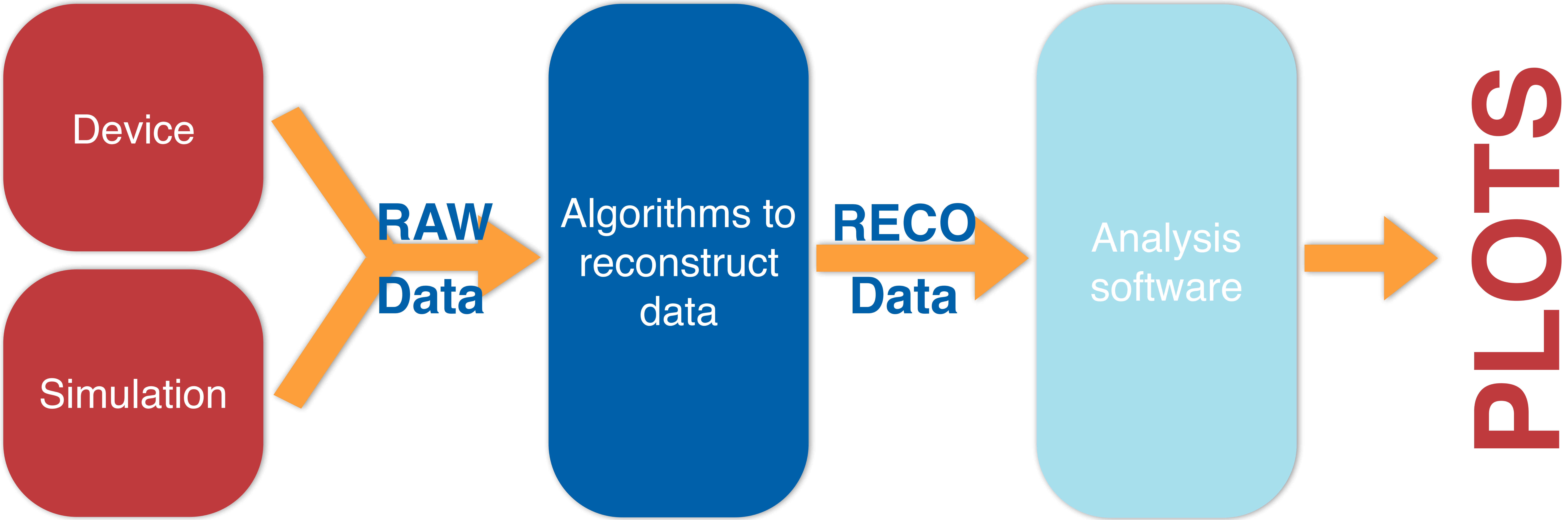


- LHC collaborations are measured in **thousands** of physicists
- Collaborations are investigating many topics in parallel and release **~hundred** publications per year
- Each analysis topic needs to look at the data in its individual way

The Scientific Process in HEP



Software & Computing

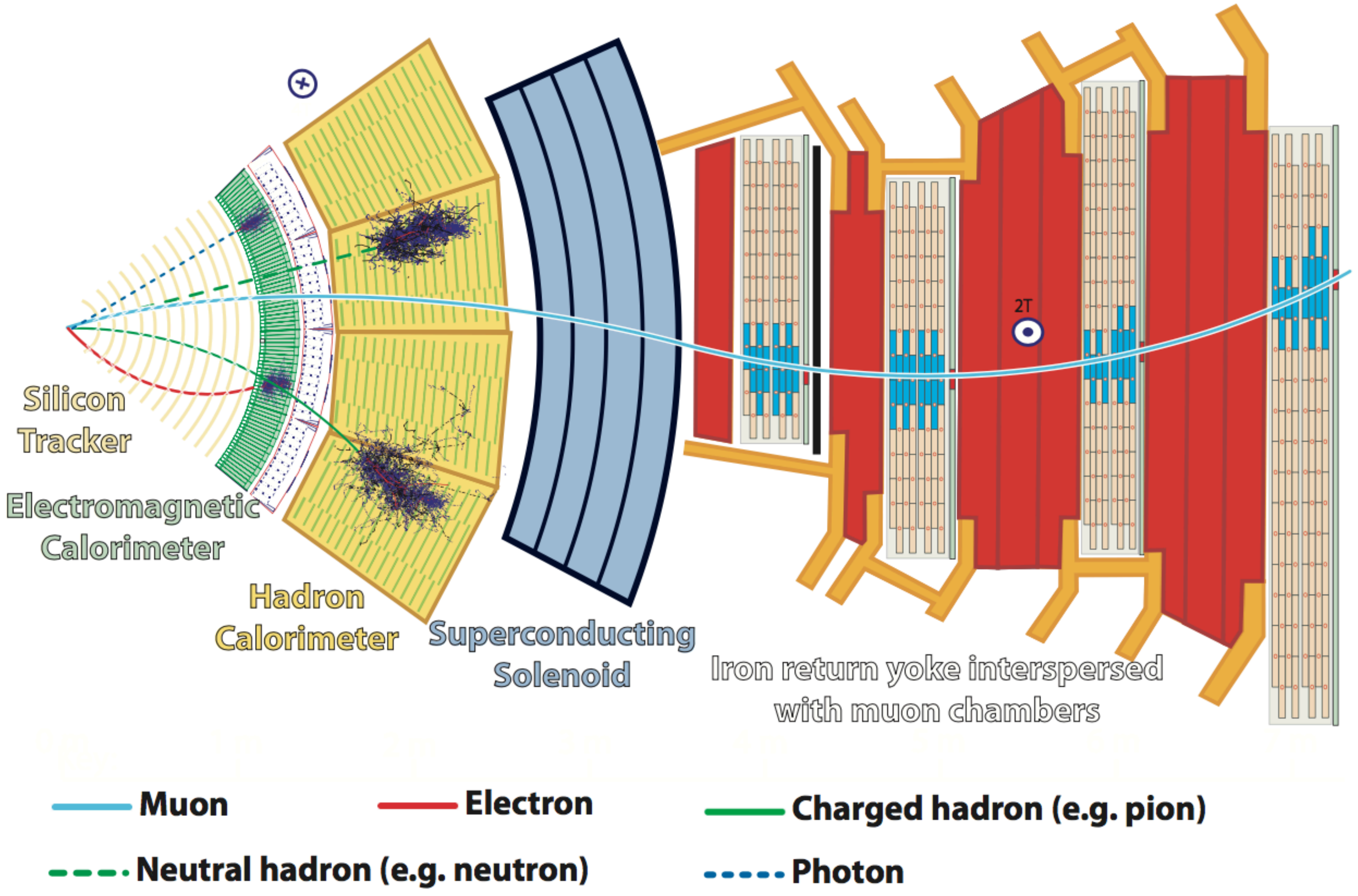


- Detector signals (and equivalent simulated signals) need to be reconstructed to learn about the particles that produced them

Software & Computing

- **Central:** organized processing/production, one software stack/framework per experiment (C++), one or few output sets, shared by large parts of the experiments for analysis
- **Chaotic:** smaller groups down to individuals explore the data for analysis, individually implemented analysis code, because of data volumes requires data reduction to keep notion of interactivity

HEP event processing application



- Reconstructed event properties are used for analysis

- HEP computing problem: HTC (High Throughput Computing)
  - Trivially parallelizable
- Individual applications access input data and produce output data
  - Input never never changed
  - New information is generated and stored in output
  - Input information copied forward or discarded
- Highest efficiency application: single thread executable
  - Nowadays memory limits per core require multi-threaded applications (4-8 standard)

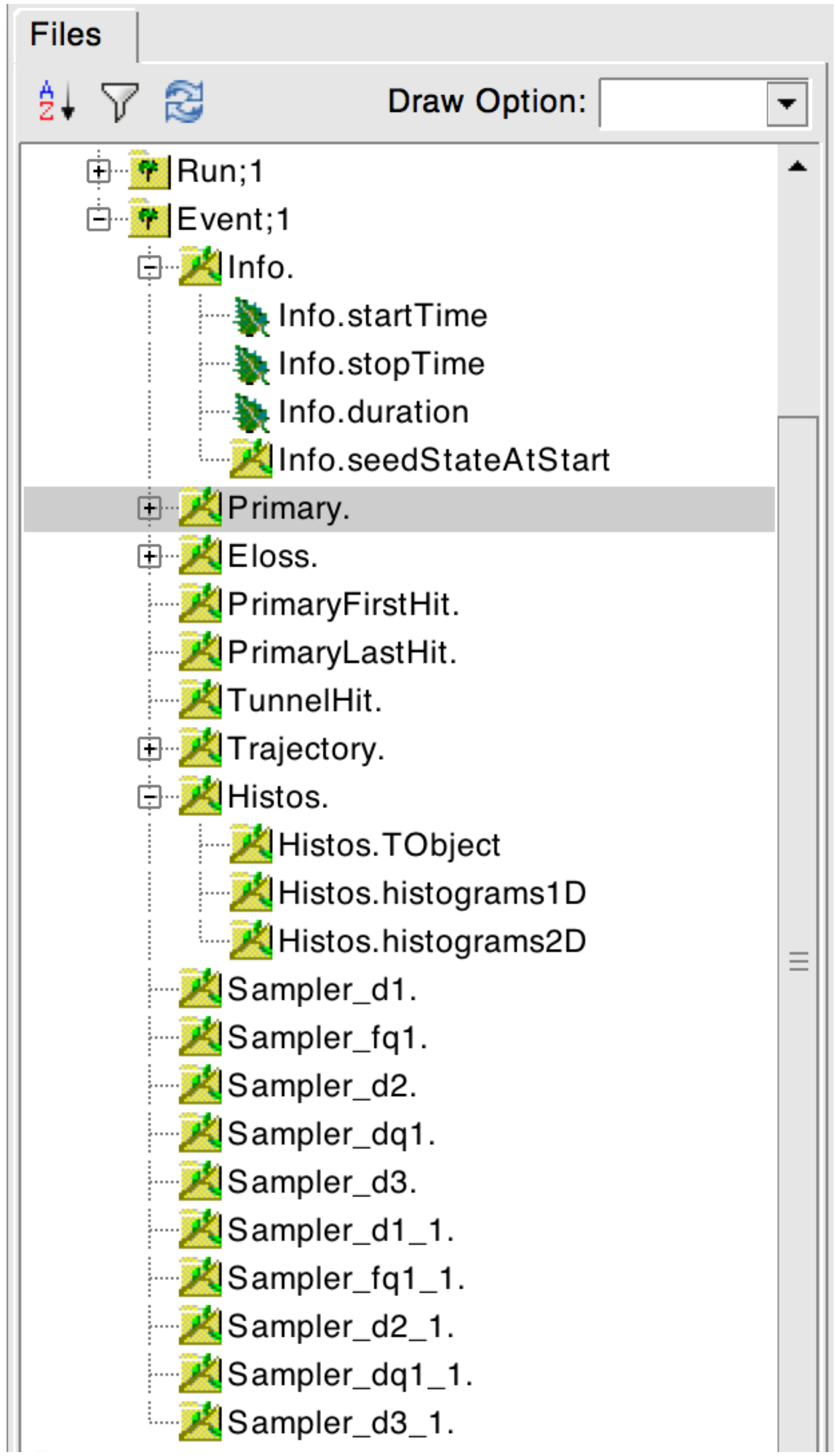
The files and file content: The Event Data Model (EDM)

- Software: Object-Oriented Framework
  - Data stored as objects in memory

- EDM: C++ type-safe container called edm::Event.
  - Any C++ class can be placed in an Event, there is no requirement on inheritance from a common base class
  - Holds all data of an event, RAW or simulated data and reconstructed data products
  - Contains metadata describing
    - Configuration of the software
    - Conditions and calibration data identification

- The Event data is output to files browsable by ROOT. The event can be analyzed with ROOT and used as an n-tuple for final analysis.

- Products in an Event are stored in separate containers
  - Collect particular types of data separately
    - particle containers (one per particle), hit containers (one per subdetector), etc.



https://root.cern.ch/input-and-output

Data Organization and Workflows

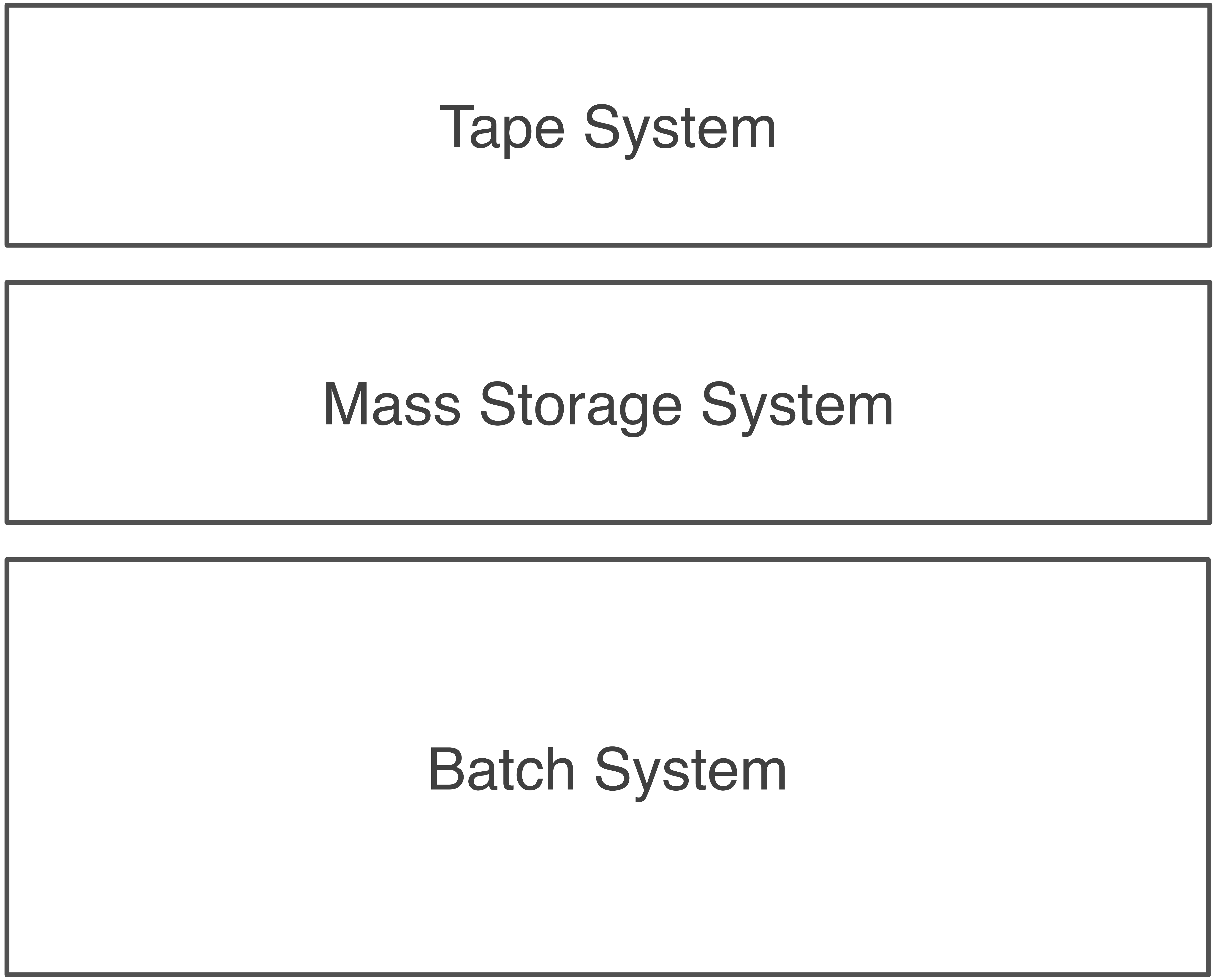
- Data Workflows:
  - Reconstruction: input RAW, output AOD/MINIAOD
  - produce MINIAOD: input AOD, output MINIAOD
  - AOD analysis: input AOD, output NTuple
  - MINIAOD analysis: input MINIAOD, output NTuple
  - User Analysis: input NTuple, output plots, tables
- Simulation Workflows
  - Generation: input nothing, output GEN
  - Simulation: input GEN, output GEN-SIM
  - Digitization/Reconstruction: input GEN-SIM, output AODSIM/MINIAODSIM
    - Digitization reads one to hundreds of additional events from secondary files
  - produce MINIAODSIM: input AODSIM, output MINIAODSIM
  - AODSIM analysis: input AODSIM, output NTuple
  - MINIAODSIM analysis: input MINIAODSIM, output NTuple
  - User Analysis: input NTuple, output plots, tables

- Data Tiers
  - RAW: RAW detector data
  - RECO: Reconstructed Information
  - AOD: Analysis Object Data, subset of RECO
  - MINIAOD: Slimmed subset of AOD
- Simulation Data Tiers
  - GEN: event generator
  - GEN-SIM: simulated information
  - GEN-SIM-RECO: reconstructed information of simulation
  - AODSIM
  - MINIAODSIM
- Special files with flat structure (no classes)
  - NTuples (favorite for analysis)

Computing Infrastructure for CMS

- Over 70 sites world-wide
  - ~150,000 cores
  - ~75 Petabyte Disk
  - ~100 PB used tape space

High Throughput Computing - Local Computing Resources



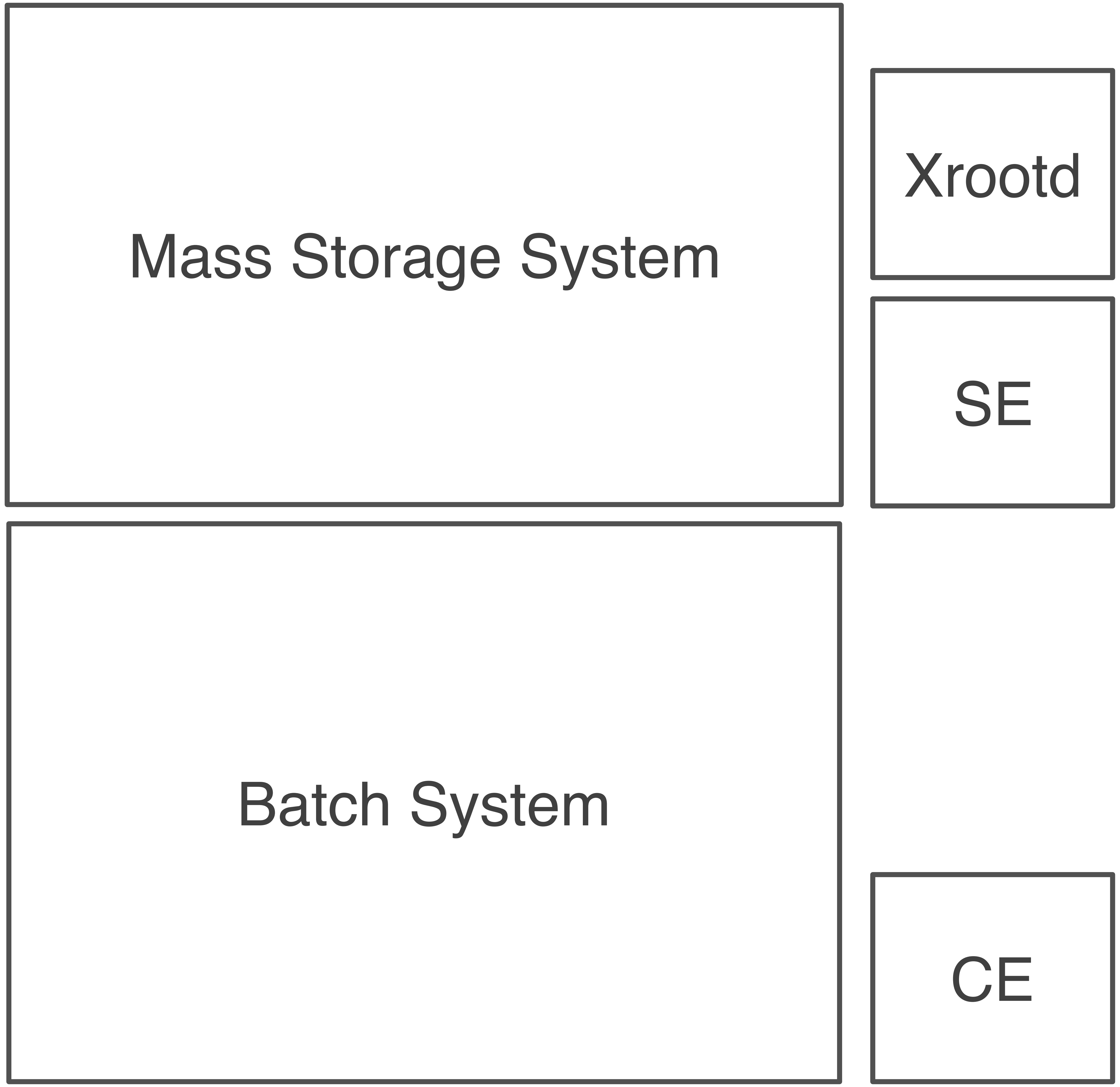
- Applications get scheduled by batch system in parallel on available resources
- Access data from Mass Storage System
  - EOS, dCache, HDFS, Lustre, Ceph, …
- Output is written back to Mass Storage System
- Tape is used for longterm archival

The sites, the transfers and the access methods

- Files are grouped in datasets
  - Datasets have same physics content (trigger or simulated collision type)
- Datasets are distributed across disk at all the sites automatically
  - Balanced and replicated according to popularity (the more in demand a dataset is, the more popular it is, the more it is replicated at different sites) ➜ more resources for more popular datasets
- Jobs are sent to sites which hold input dataset

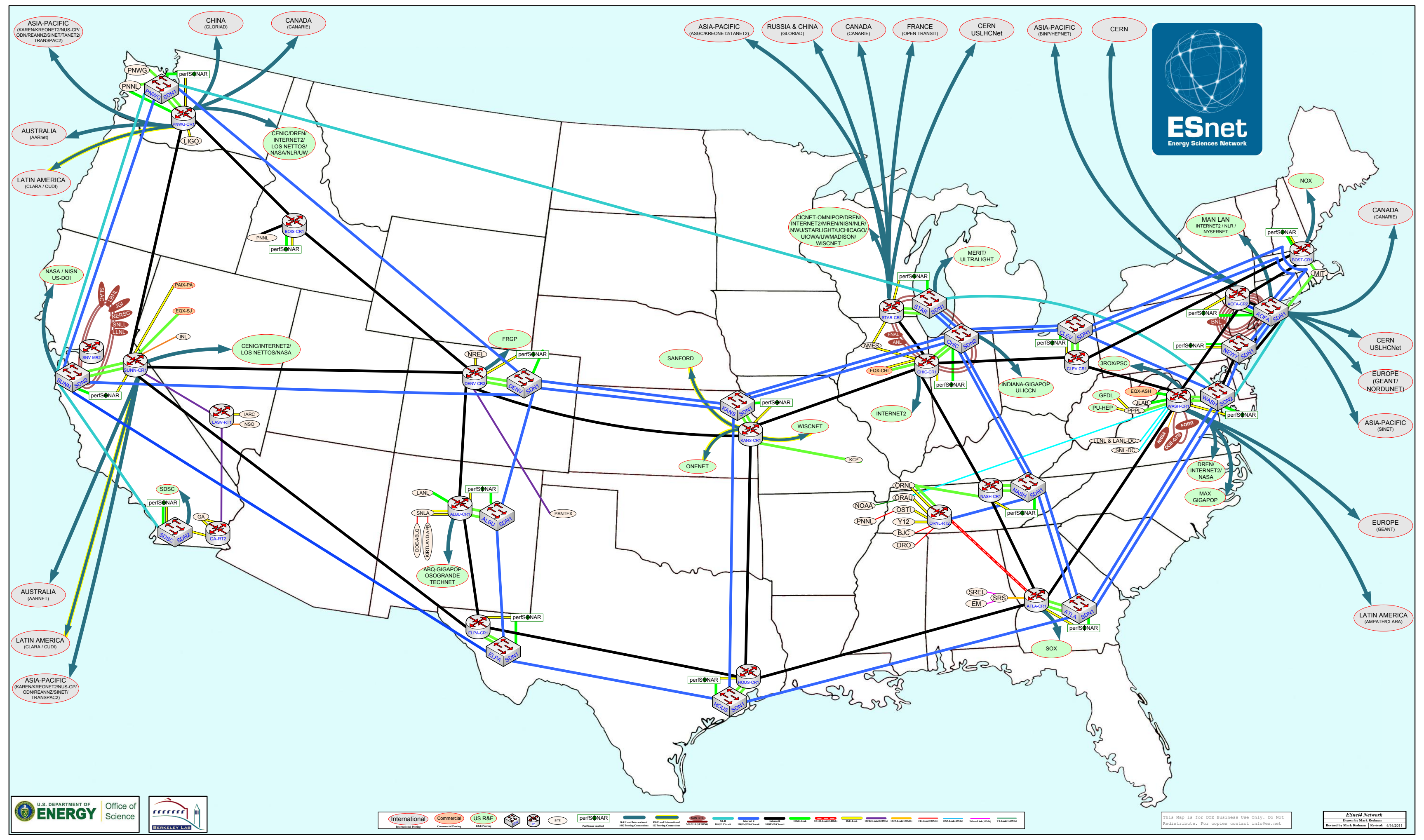
- Additional access method: data federation
  - xrootd based data federation can discover files stored at all sites and stream file content over WAN (EDM optimized for WAN access with file content caching, etc.)

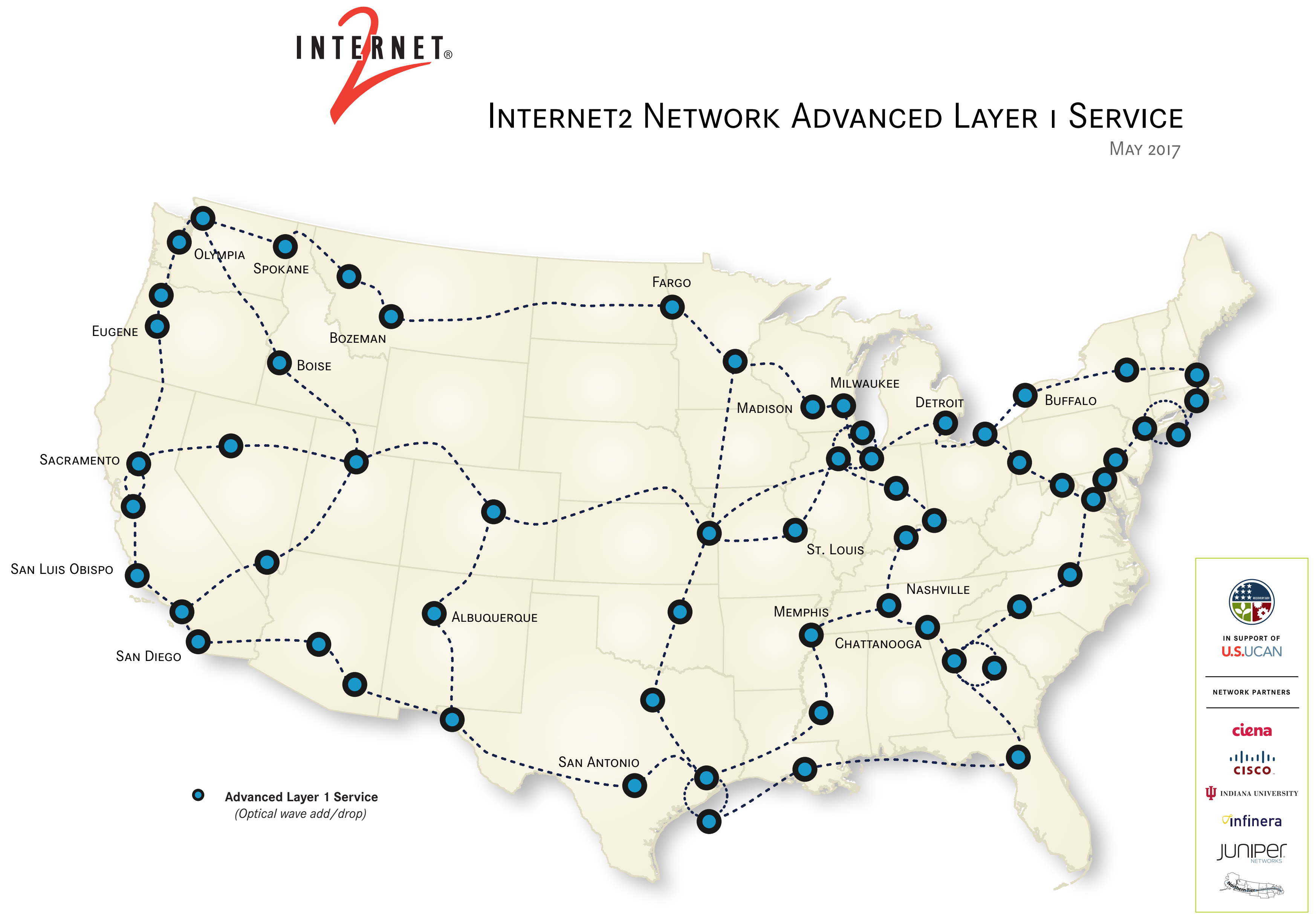
Network access to distributed computing resources

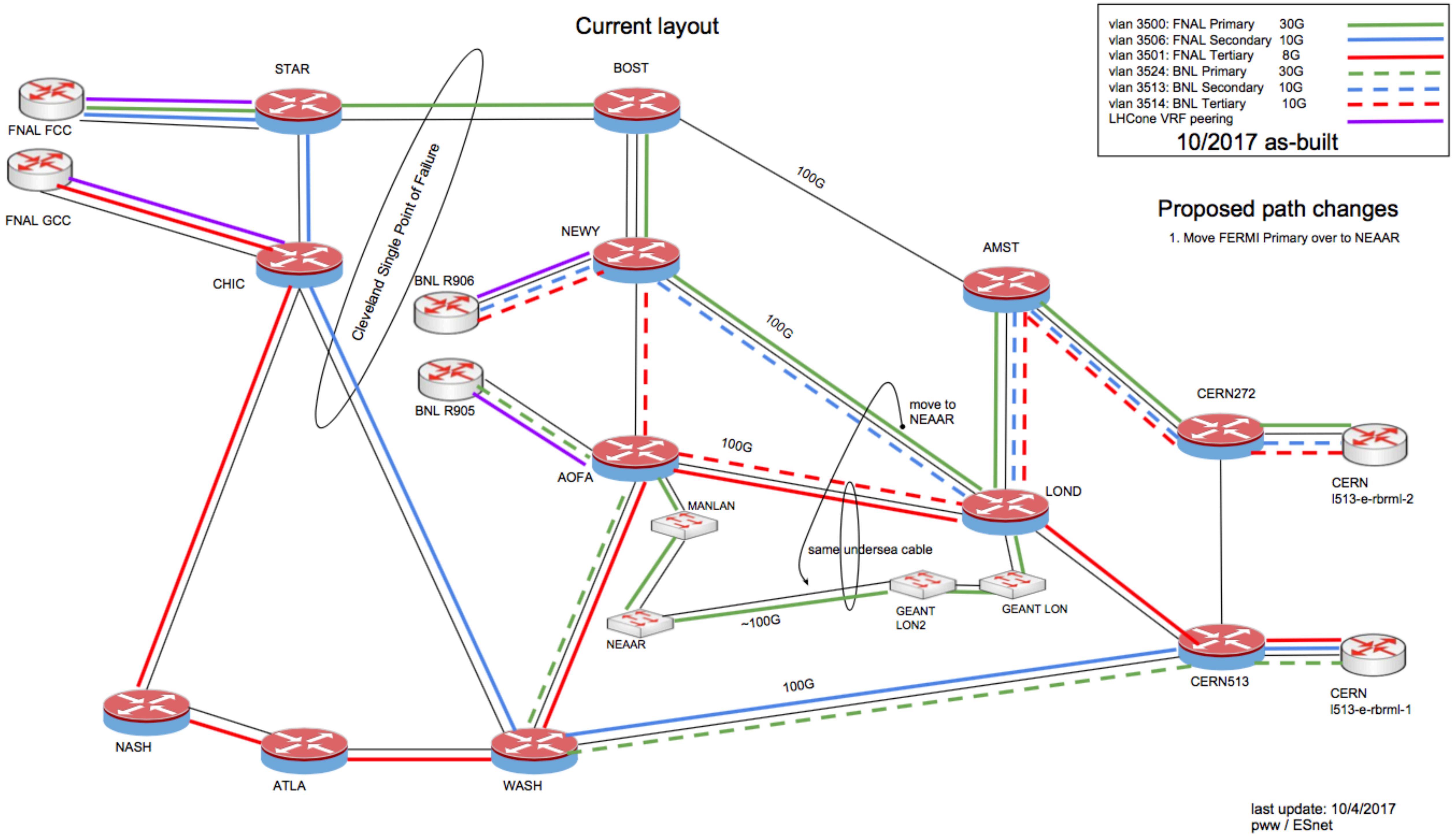


- CE: Compute Element
  - X509 certificate authenticated access to batch system
- SE: Storage Element
  - X509 certificate authenticated access to storage system (copy to site and retrieval from site)
- Xrootd: streaming service
  - Direct access to files stored at site from remote applications

Networks

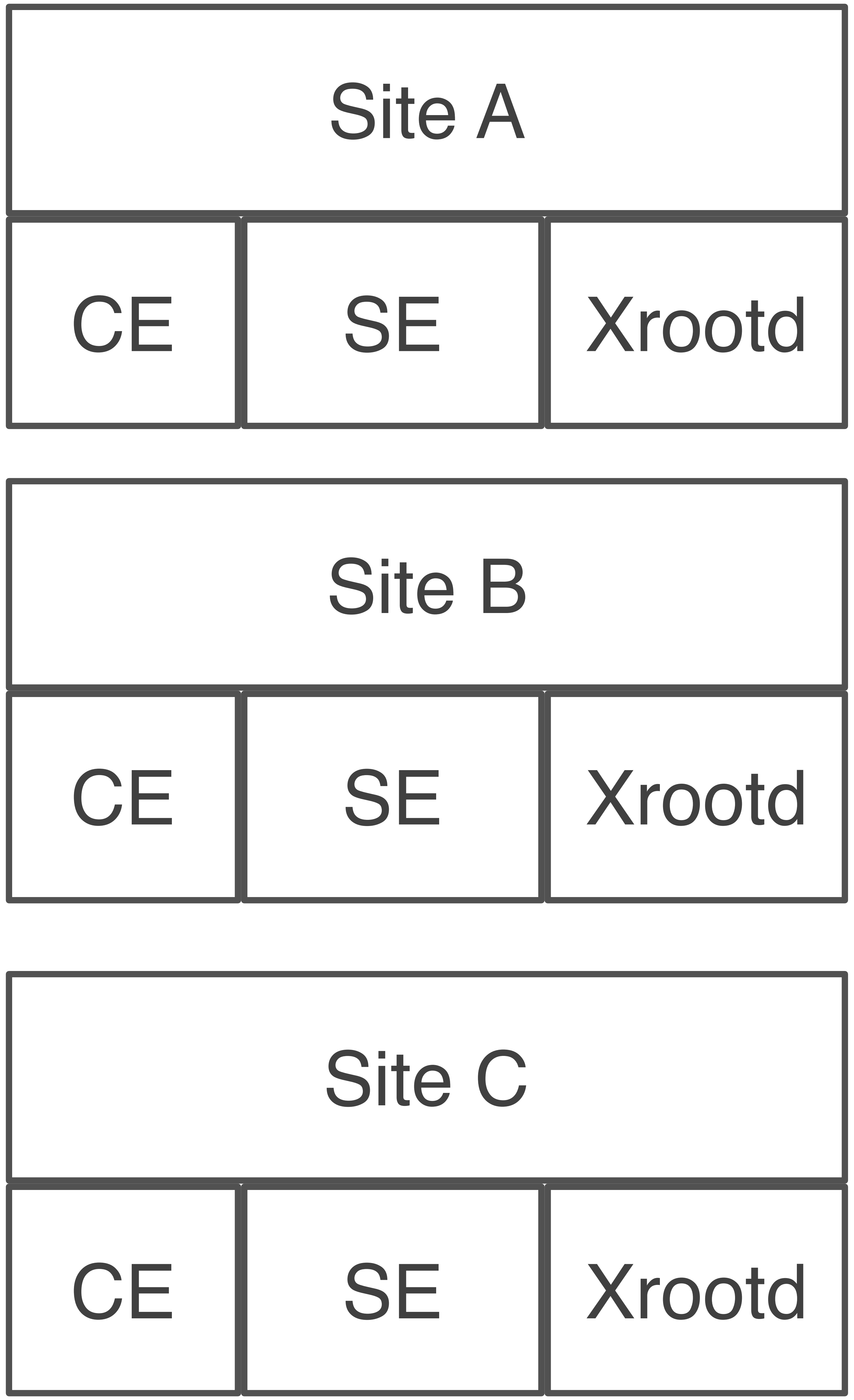






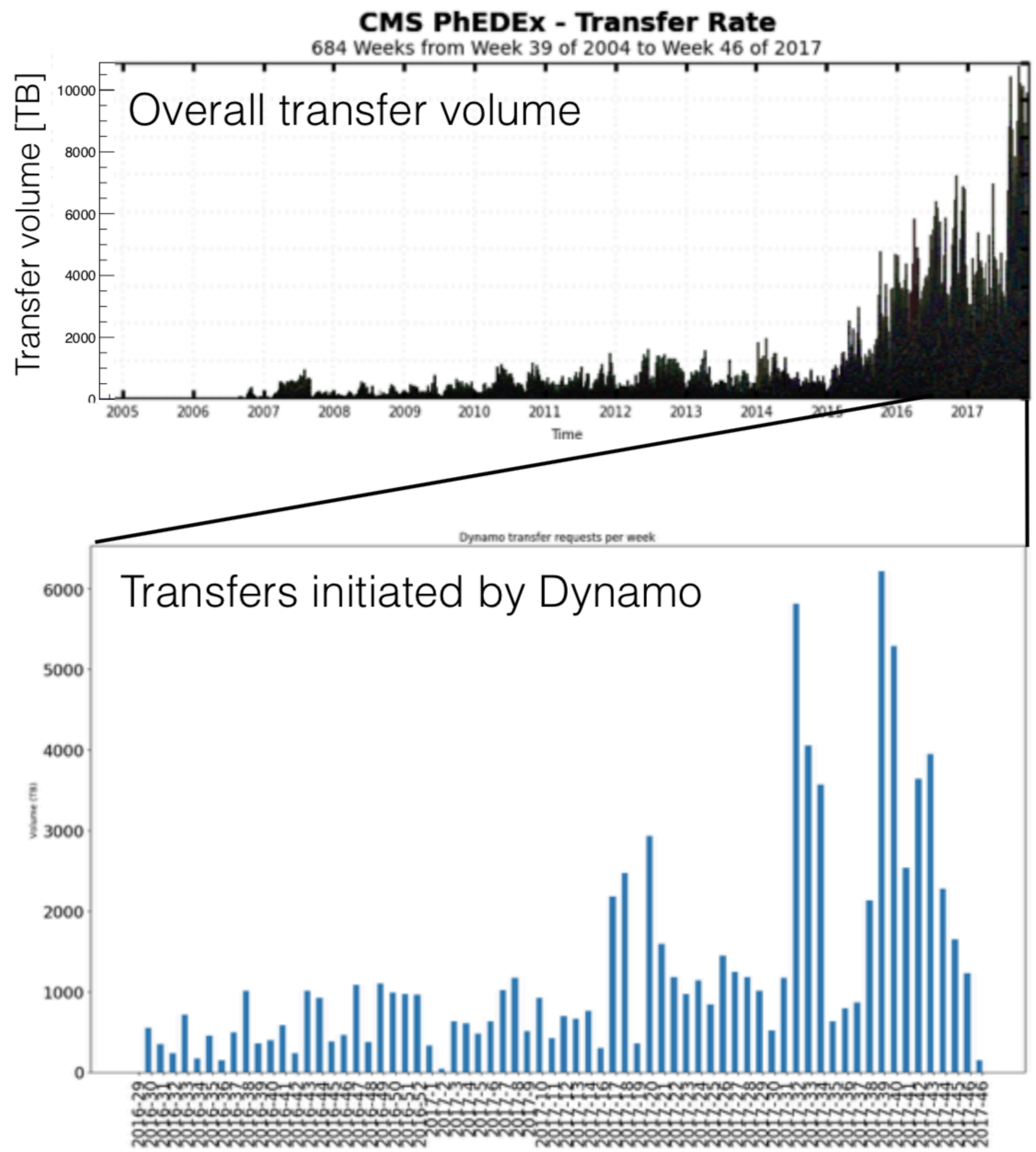
- In the US, sites are connected with 100 Gbps through the science networks ESNet and Internet2
- Transatlantic capacity: 340 Gbps

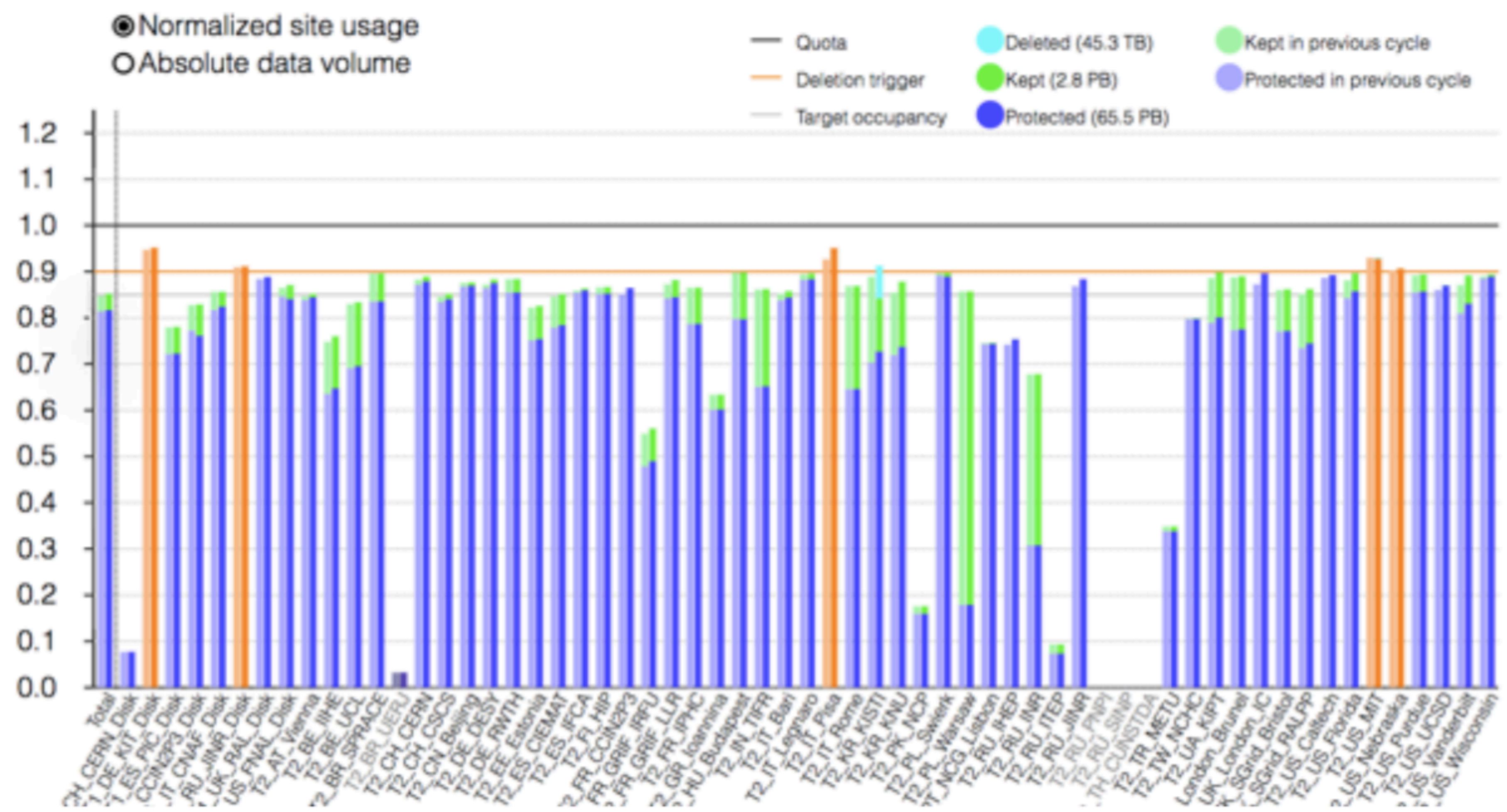
Distributed Infrastructure



- Transfer system
  - Tracks file locations at sites (replica db)
  - Initiates transfers of files between sites
- FTS
  - Handles the actual file transfers
  - Uses protocols like GridFTP and others
  - Has queueing, bandwidth scheduling and retry logics
- Metadata catalog
  - Stores metadata of files
- Query system
  - Access to replica db and metadata catalog
  - Place to find information and location of data (both for users and automated systems)

CMS Transfer system - Dynamic Data Management

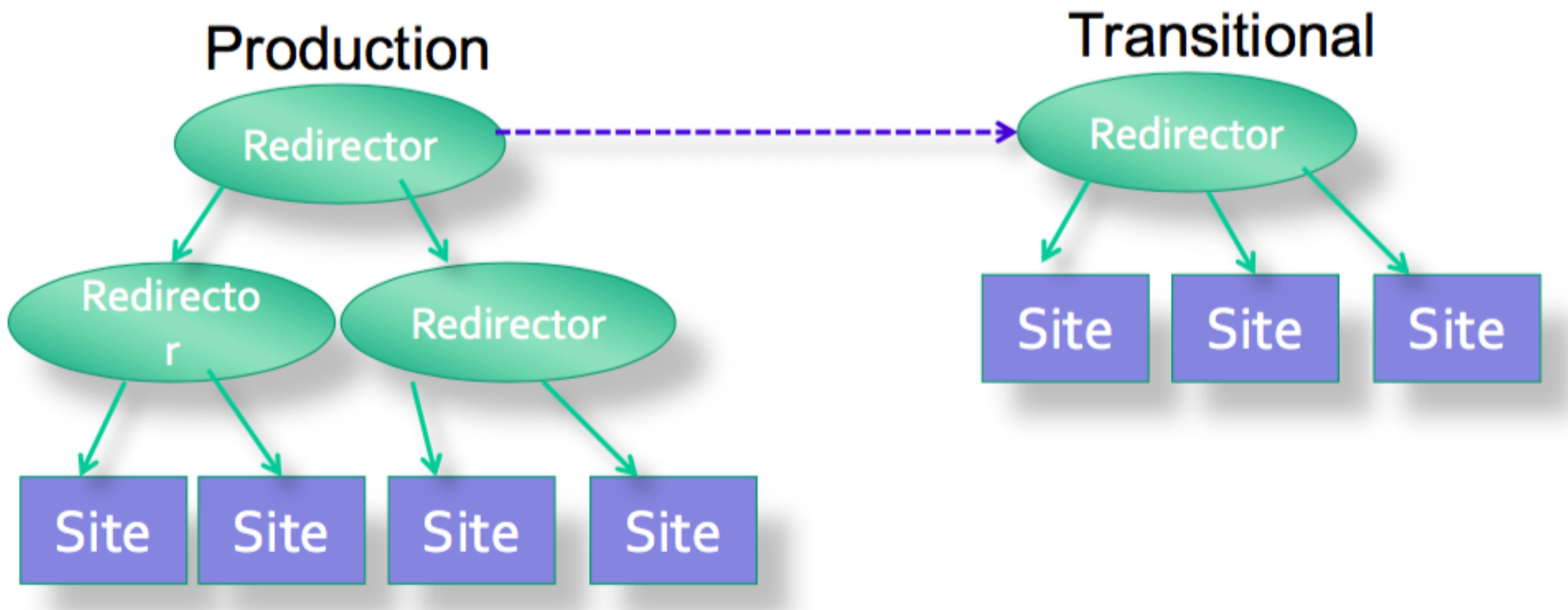


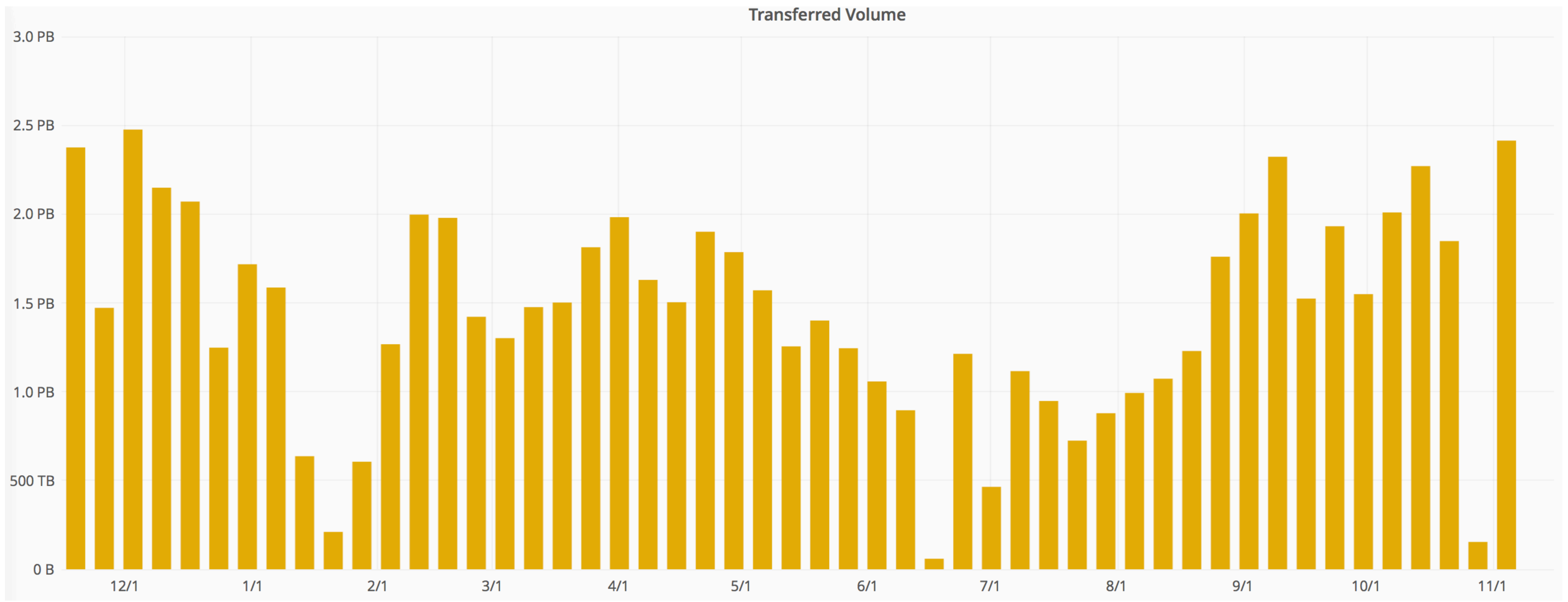


- Disk space dynamically managed according to popularity of data
  - More replicas to provide access to more applications in parallel
- Currently: ~10 PB per week

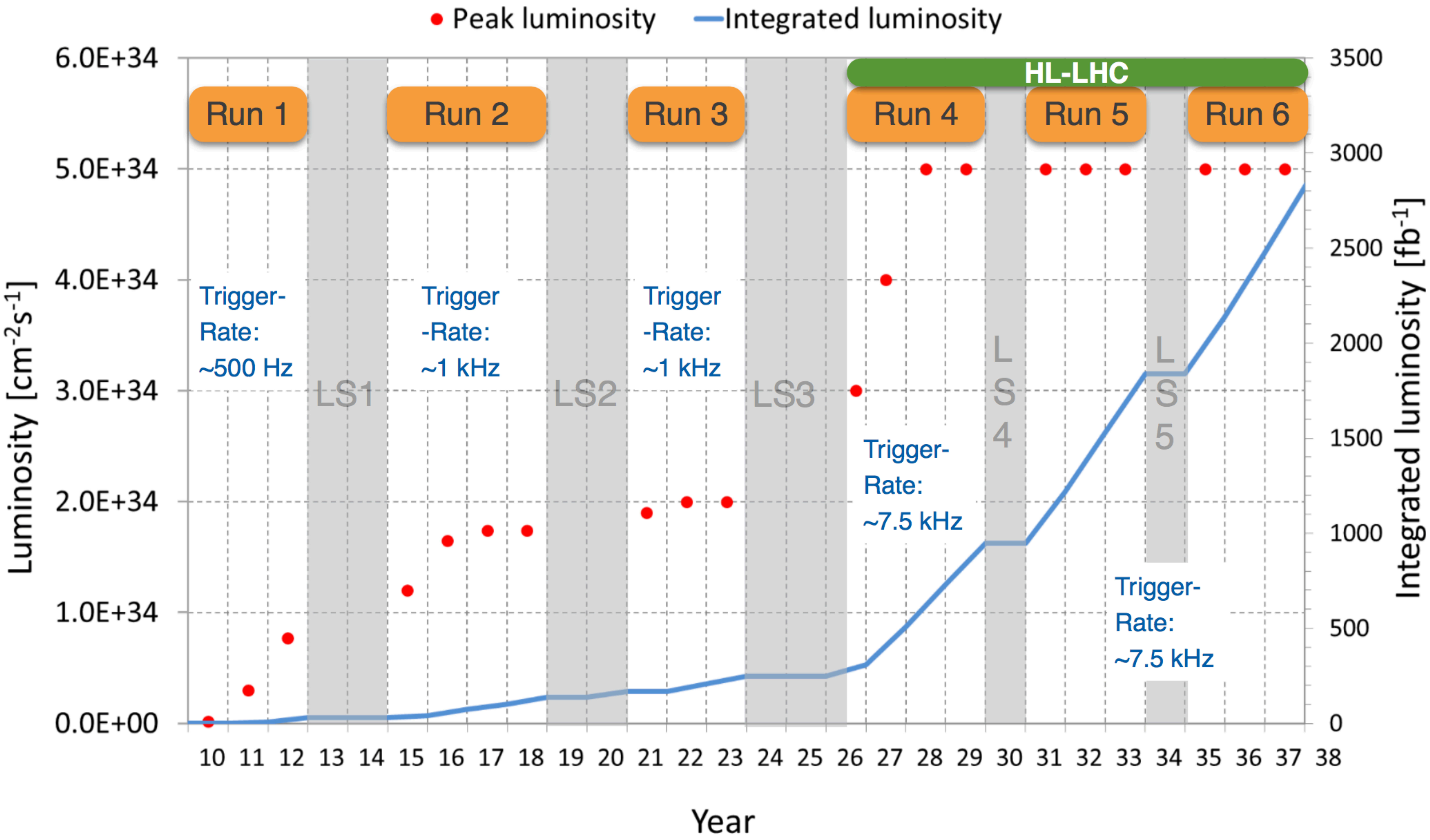
Xrootd: CMS Any Data, Any Time, Anywhere

- Based on opening a file through the wide area network
  - Example: file is stored at CERN but the application opens the file at Fermilab
  - Latency is significantly reduced (almost unnoticeable) through clever caching and pre-fetching in CMSSW
  - Works transparently if the requested file is not available locally
- Currently: ~2 PB per week

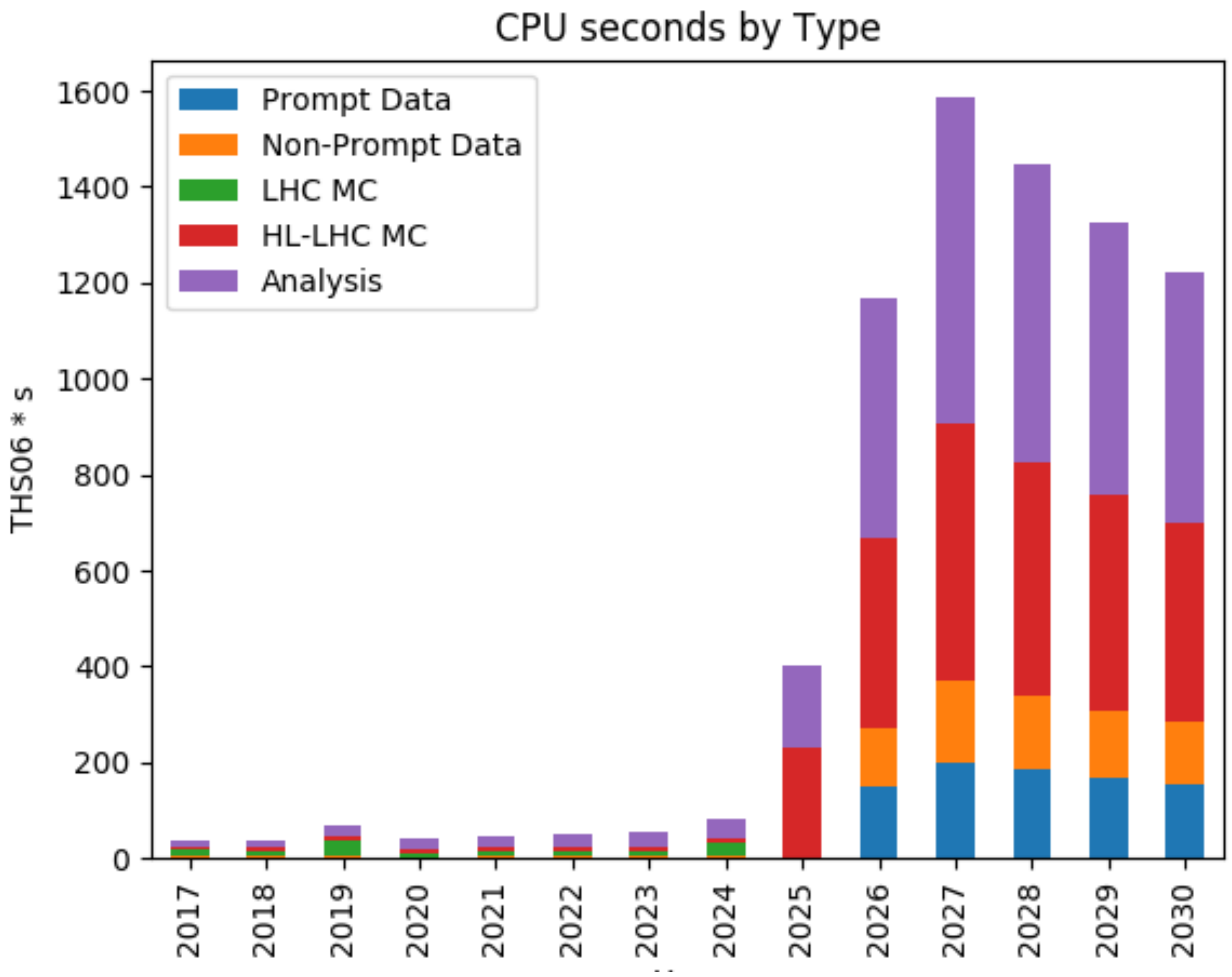


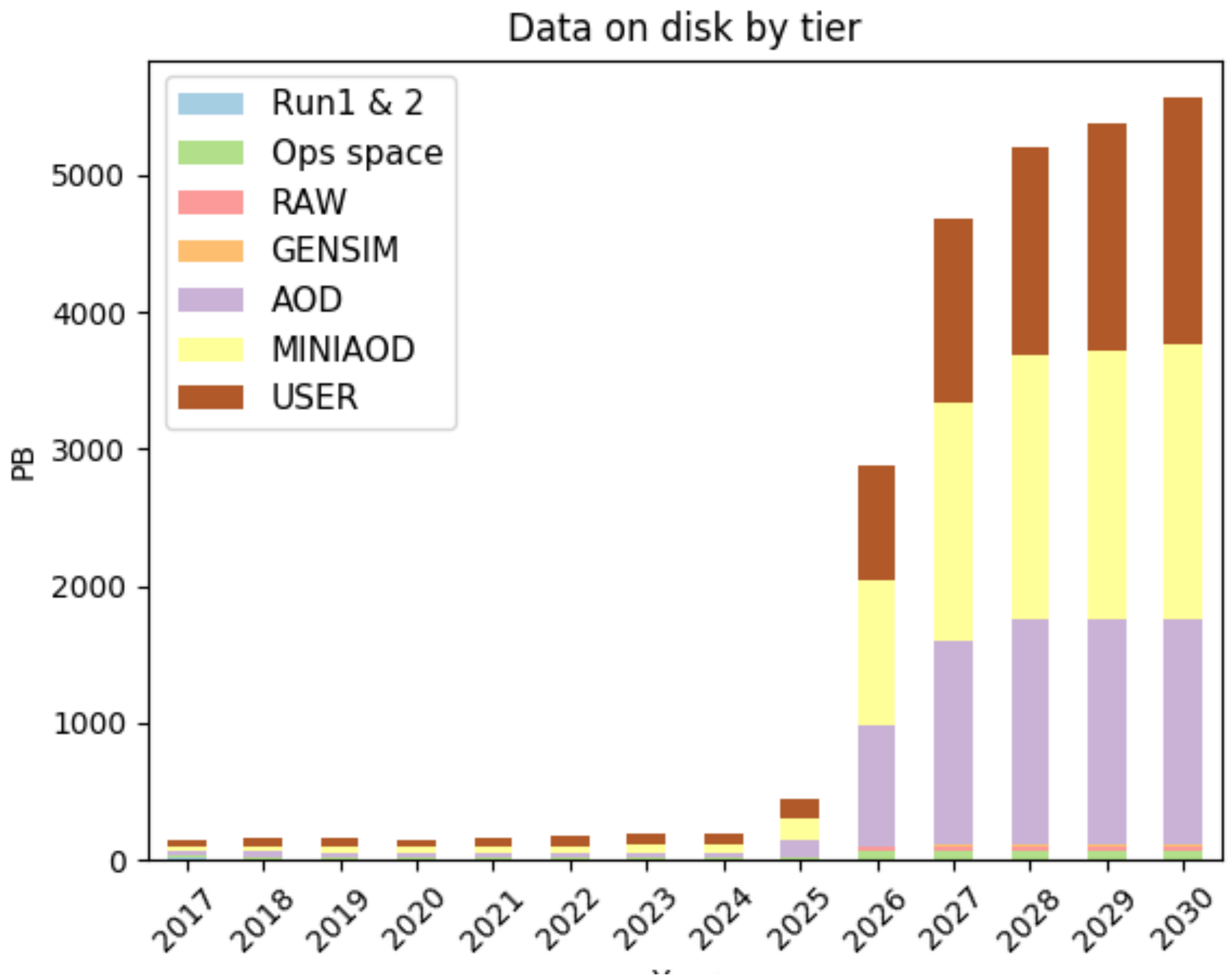


HL-LHC


Peak luminosity
Integrated luminosity
Luminosity [cm-2s-1]
Integrated luminosity [fb-1]
Year
HL-LHC
Run 1
Run 2
Run 3
Run 4
Run 5
Run 6
LS1
LS2
LS3
LS4
LS5
Trigger-Rate: ~500 Hz
Trigger-Rate: ~1 kHz
Trigger-Rate: ~1 kHz
Trigger-Rate: ~7.5 kHz
Trigger-Rate: ~7.5 kHz
6.0E+34
5.0E+34
4.0E+34
3.0E+34
2.0E+34
1.0E+34
0.0E+00
3500
3000
2500
2000
1500
1000
500
0
10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38

HL-LHC: Naive Extrapolation of current computing model





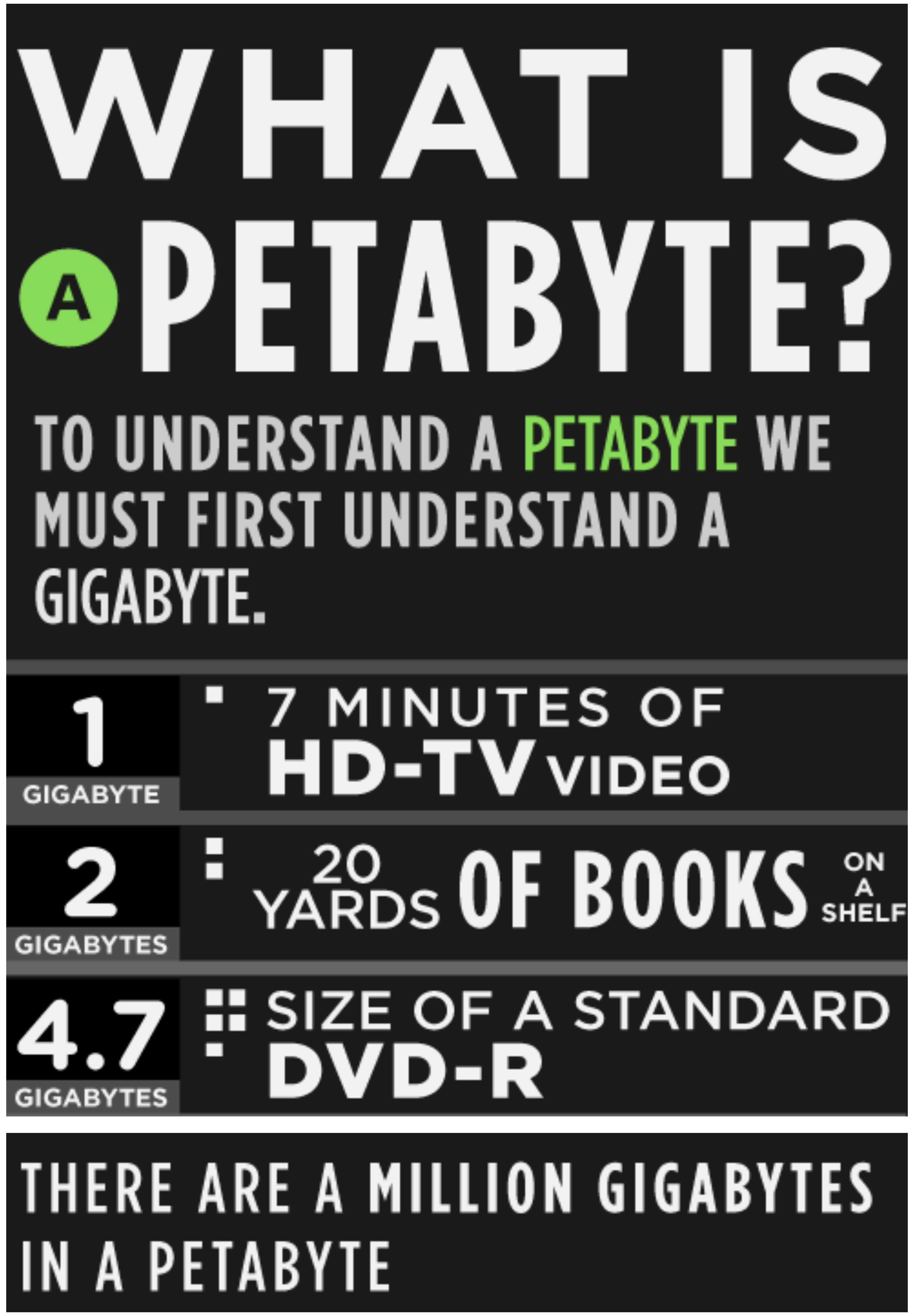
2027: **5 Million cores** for 1 year

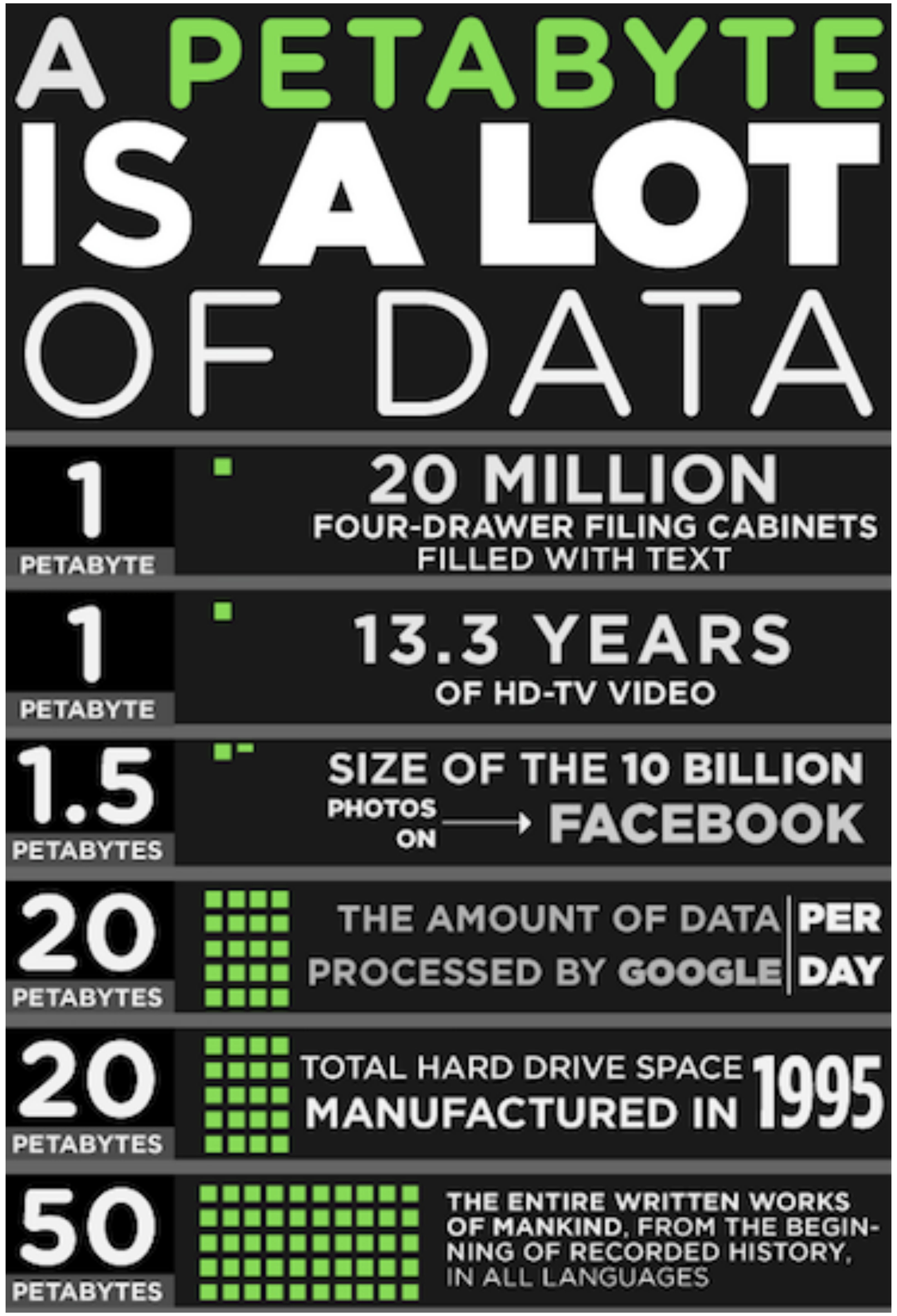
2027: **5 Exabytes of disk**


Fermilab
U.S. DEPARTMENT OF ENERGY
Office of Science

Backup

What is a Petabyte?





Backup: Data Organization

- Event
  - real data (collision events)
  - MC events
- Lumi section
  - 23 seconds of data taking
  - Variable number of events for MC
    - number is calculated to be able to digitize-reconstruct all events in a lumi section in 12-24 hours
- Run
  - Run number from during data taking
  - MC has arbitrary run number
- Every data event is uniquely identifiable by
  - RUN:LUMI:EVENTNR
- Files
  - only complete lumi sections
  - necessary for correct book keeping
  - all uniquely named (with random name)
- Blocks
  - group of files
  - designed to fit on a tape cartridge to optimize tape staging (tape mount overhead and searching on tape is significant)
- Dataset
  - Group of blocks

Backup: Dataset name

- /<primary dataset name>/<era>-<processed dataset name>-<version>/<data tier>
  - Block: <dataset name>#<uuid (unique identifier>

- primary dataset name
  - Data: Trigger selection (SingleMuon, MultiJet, …)
    - Intention: analyses will only have to access one or a limited number of primary datasets ➜ reduces amount of computing resources needed for analysis
  - Overlap of 10-15% is included in the planning of computing resources
  - MC: generated process (simple explanation, can be more complicated)

- era
  - Groups data with similar data taking conditions together
    - for example, when the machine parameter change, we change the era (Run2016A-B-C ...)
  - MC era corresponds to MC production campaign name

- processed dataset name
  - separates different processings of the same data (prompt, re-reconstruction, skims)

- version number
  - Distinguish different processing passes or small changes (for example if the data tier content changes although the data taking conditions stay the same)

- data tier name
  - Defined groups of persisted objects (event content) from processing steps: RAW, RECO, AOD, MINIAOD, GEN-SIM, GEN-SIM-RECO, AODSIM, MINIAODSIM
  - changes in event content need a new CMSSW release
  - Individual datasets have only one data tier and event content ➜ enables processing and analysis execution of same code (CMSSW version + module schedule) per dataset

Backup: The book keeping problem

- Data storage atomic unit: block

- The problem: How do we keep track of blocks at all the CMS sites:
  - Every site has their own storage setup and has different file paths (like a mount point in your linux distribution)
    - Site A: /mnt/eos/my-directory/my-root-file
    - Site B: /big-storage-system/my-other-directory/my-other-root-file
    - This is called the **Physical File Name (PFN)**
  - CMS created its own namespace which is the same at all sites
    - maps the file of a dataset into a **Logical File Name (LFN)**
    - /store/data/<era>/<primary dataset name>/<data tier name>/<processed dataset name>-<version>/<counter>/<filename>
  - Every site has a mapping between LFN and PFN
    - $CMS_SITE/SITECONF/local/PhEDEx/storage.xml
    - CMSSW and other systems can use these translations automatically
      - specifying an LFN in a CMSSW job is sufficient to open a file
  - **This is the reason why CMS data storage and distribution scales to many sites!**